ΔΕΣΜΗ 2009-2010

Research Promotion Foundation

ΚΥΠΡΙΑΚΗ ΔΗΜΟΚΡΑΤΙΑ

Achieving Reliability in Master-Worker Computing via Evolutionary Dynamics

Evgenia Christoforou — University of Cyprus

Antonio Fernandez Anta — Institute IMDEA Networks

Chryssis Georgiou — University of Cyprus

Miguel Mosteiro — Kean University and URJC

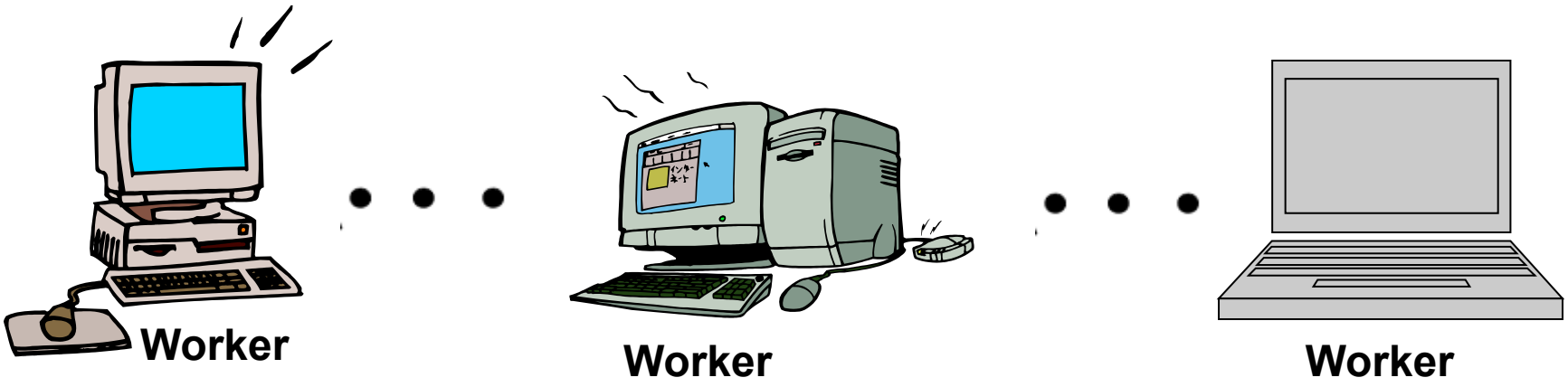
Angel Sanchez — Univ. Carlos III de Madrid

July 17th, 2012

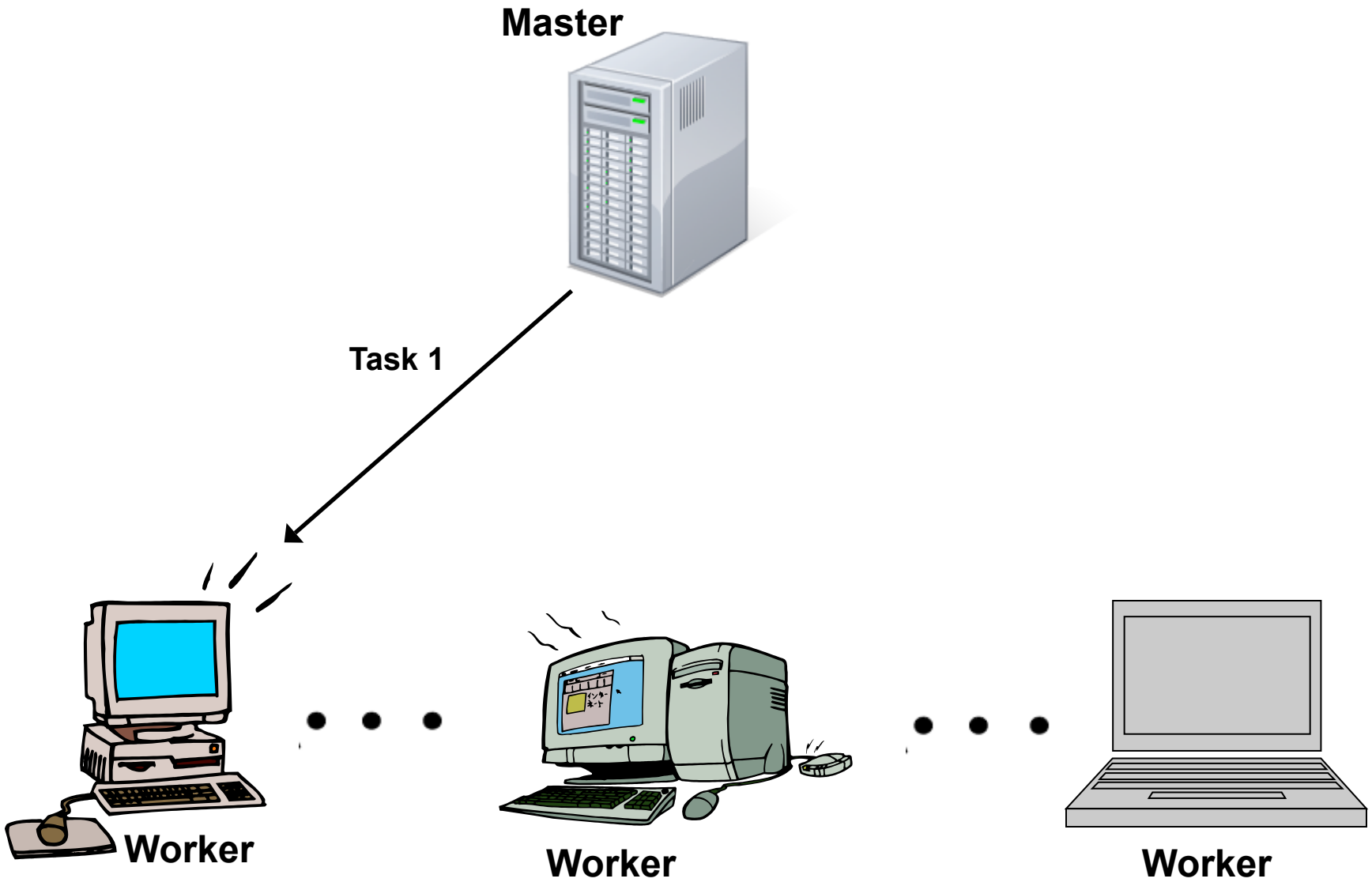
Madeira, Portugal

This work is supported in part by the Cyprus Research Promotion Foundation grant ΤΠΕ/ΠΛΗΡΟ/0609(ΒΕ)/05

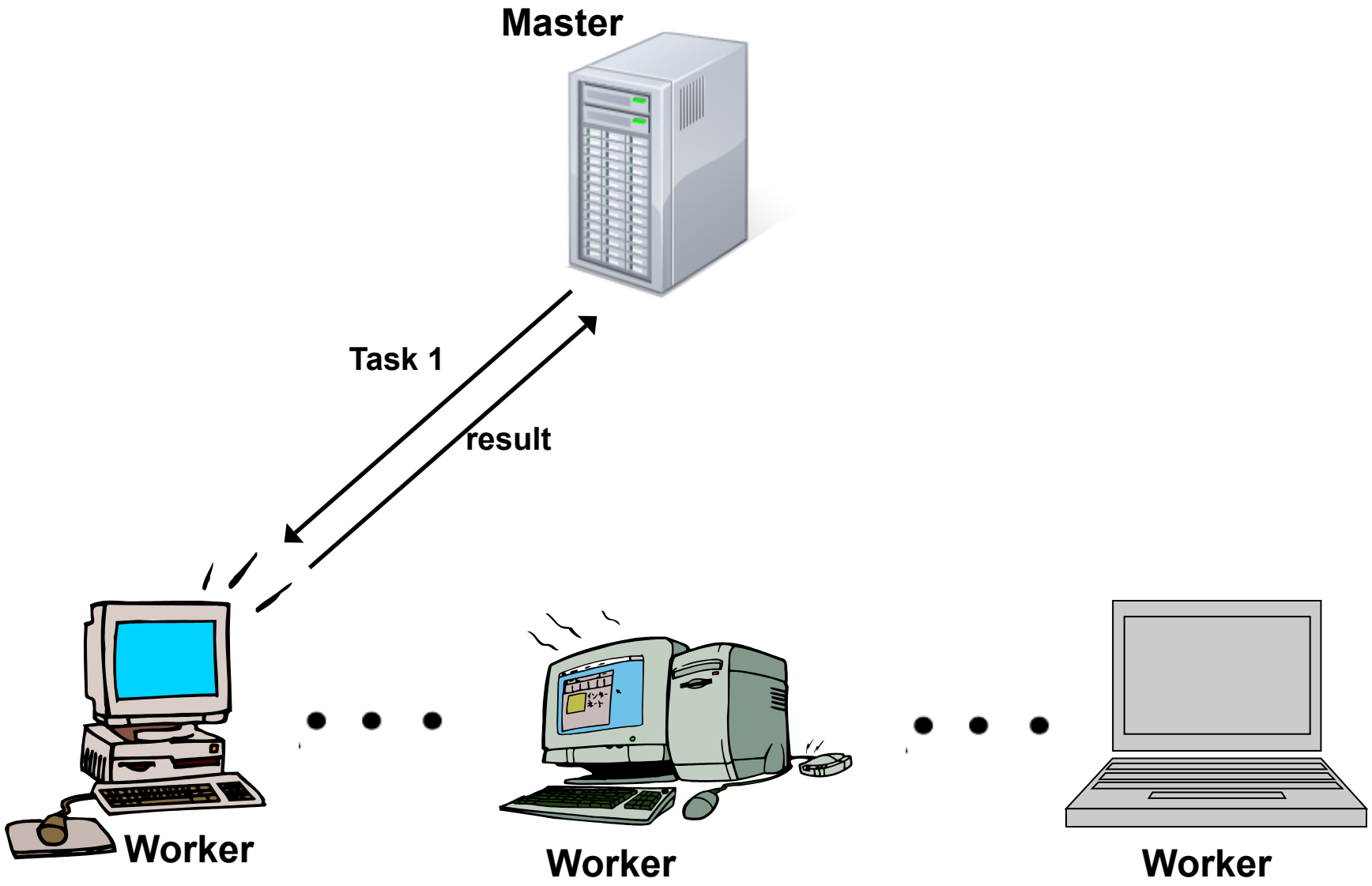
SETI-like Internet-based Computing



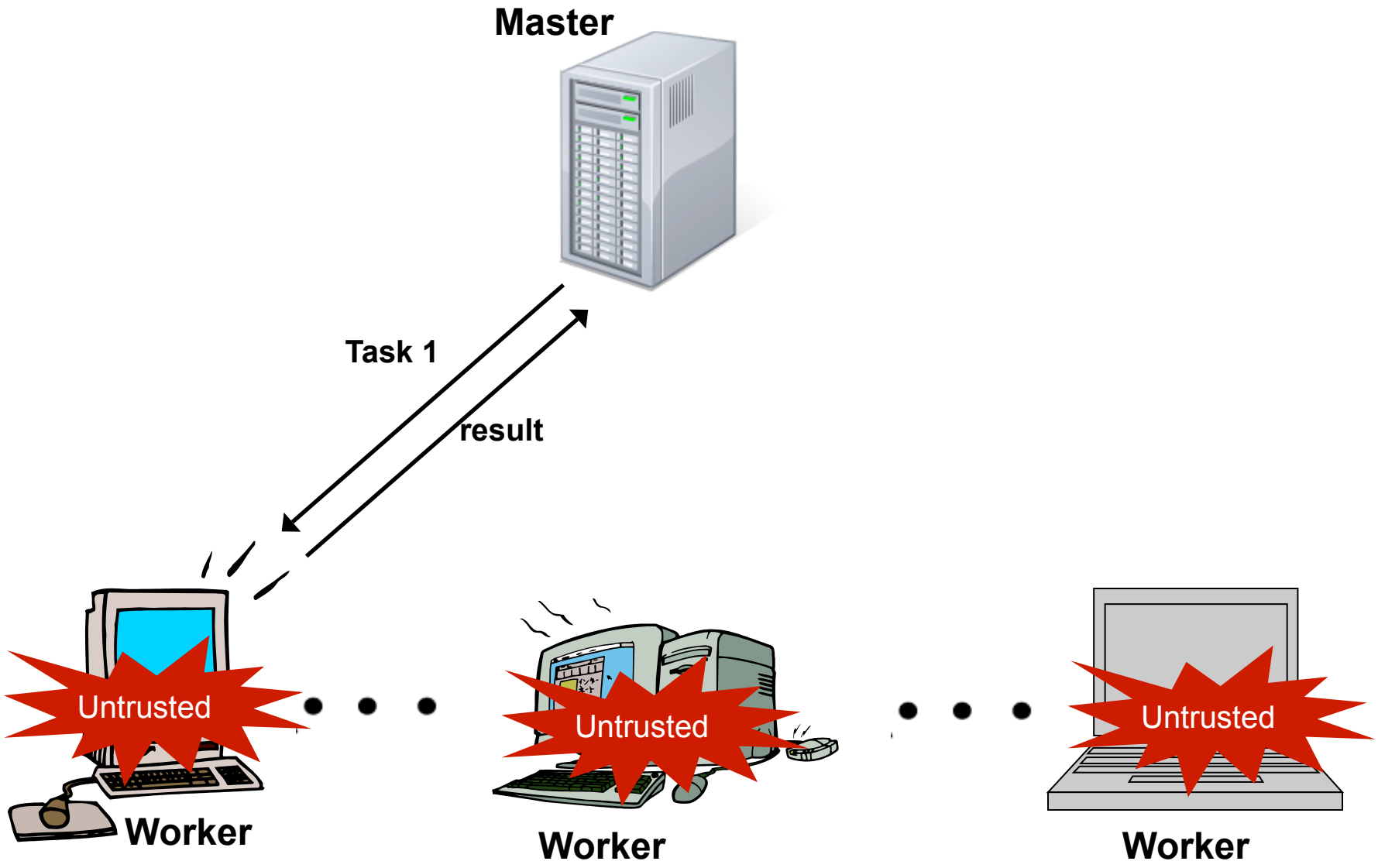
SETI-like Internet-based Computing



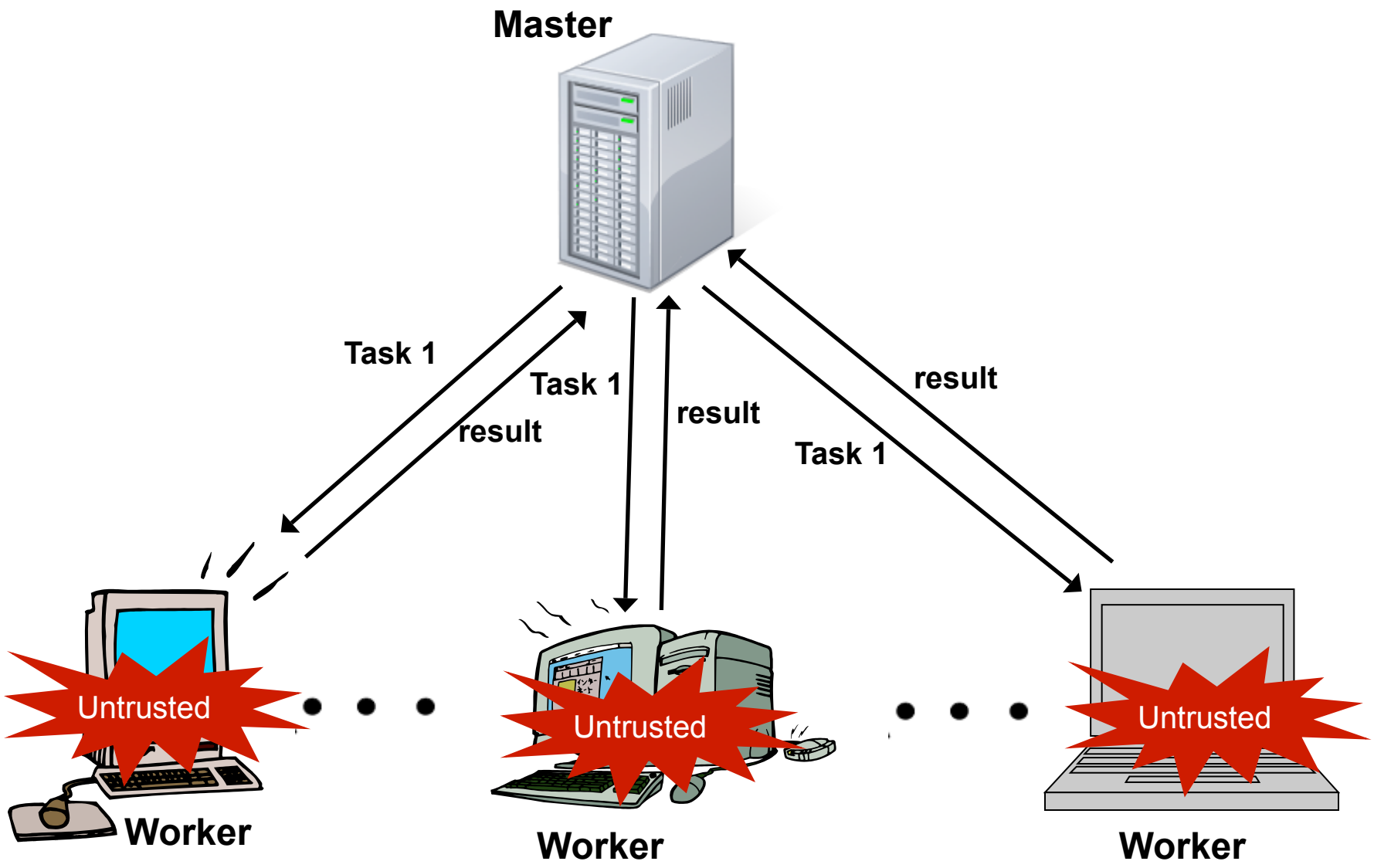
SETI-like Internet-based Computing



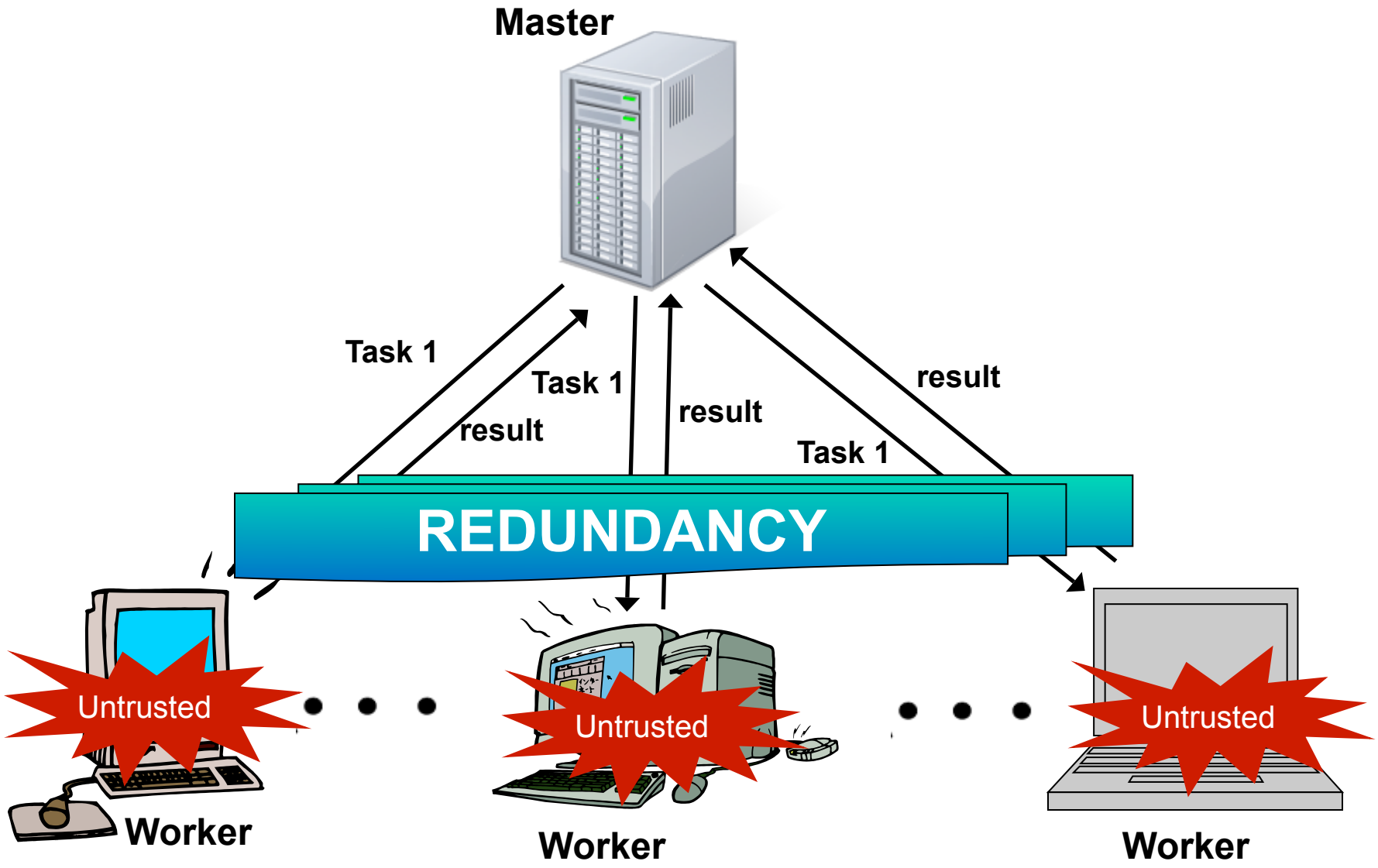
SETI-like Internet-based Computing



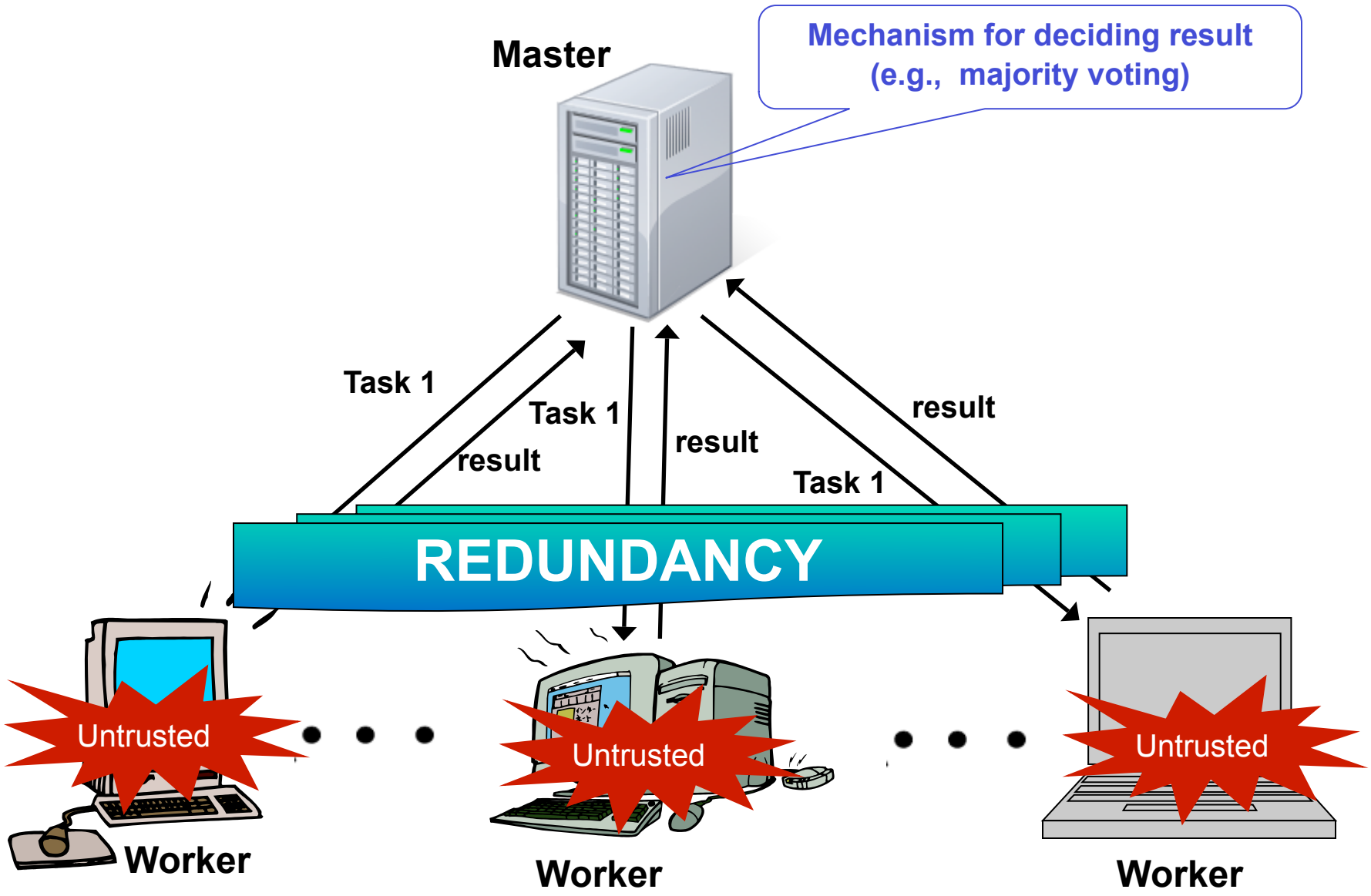
SETI-like Internet-based Computing



SETI-like Internet-based Computing



SETI-like Internet-based Computing



Prior Work

- Rational workers: act upon their best interest, i.e., choose the strategy that maximizes their own benefit
  - Honest: compute and report correct result
  - Cheat: fabricate and return a bogus result
- Mechanisms with reward/punish schemes that provide incentives to workers to be honest
  - One shot: in each round a task is performed and no knowledge is forwarded to the next round

Can the repeated interaction between the master and the workers be exploited effectively?

Prior Work

- Rational workers: act upon their best interest, i.e., choose the strategy that maximizes their own benefit [Shneidman Parkers 03]
  - Honest: compute and report correct result
  - Cheat: fabricate and return a bogus result
- Mechanisms with reward/punish schemes that provide incentives to workers to be honest
  - One shot: in each round a task is performed and no knowledge is forwarded to the next round

Can the repeated interaction between the master and the workers be exploited effectively?

Prior Work

- Rational workers: act upon their best interest, i.e., choose the strategy that maximizes their own benefit [Shneidman Parkers 03]
  - Honest: compute and report correct result
  - Cheat: fabricate and return a bogus result

Prior Work

- Rational workers: act upon their best interest, i.e., choose the strategy that maximizes their own benefit
  - Honest: compute and report correct result
  - Cheat: fabricate and return a bogus result

[Shneidman Parkers 03]

[Yurkewych et al 2005, Fernandez et al 2008]

Prior Work

- Rational workers: act upon their best interest, i.e., choose the strategy that maximizes their own benefit [Shneidman Parkers 03]
  - Honest: compute and report correct result
  - Cheat: fabricate and return a bogus result
- Mechanisms with reward/punish schemes that provide incentives to workers to be honest
  - One shot: in each round a task is performed and no knowledge is forwarded to the next round

[Yurkewych et al 2005, Fernandez et al 2008]

Prior Work

- Rational workers: act upon their best interest, i.e., choose the strategy that maximizes their own benefit [Shneidman Parkers 03]
  - Honest: compute and report correct result
  - Cheat: fabricate and return a bogus result
- Mechanisms with reward/punish schemes that provide incentives to workers to be honest
  - One shot: in each round a task is performed and no knowledge is forwarded to the next round

[Yurkewych et al 2005, Fernandez et al 2008]

Prior Work

- Rational workers: act upon their best interest, i.e., choose the strategy that maximizes their own benefit [Shneidman Parkers 03]
  - Honest: compute and report correct result
  - Cheat: fabricate and return a bogus result
- Mechanisms with reward/punish schemes that provide incentives to workers to be honest
  - One shot: in each round a task is performed and no knowledge is forwarded to the next round

[Yurkewych et al 2005, Fernandez et al 2008]

Can the repeated interaction between the master and the workers be exploited effectively?

Our Approach

Our Approach

- We introduce the concept of *evolutionary dynamics* under the biological and social perspective and relate them to Internet-based master-worker task computing

Our Approach

- We introduce the concept of *evolutionary dynamics* under the biological and social perspective and relate them to Internet-based master-worker task computing

[Camerer 03,Szepesvari 10]

Our Approach

- We introduce the concept of *evolutionary dynamics* under the biological and social perspective and relate them to Internet-based master-worker task computing

- Employ *reinforcement learning* both on Master and Workers
  - Positive payoffs increase probability of strategy just chosen
  - Negative payoffs reduce the probability
  - Knowledge only of the payoffs received, not of the strategies involved

[Camerer 03,Szepesvari 10]

Our Approach

- We introduce the concept of *evolutionary dynamics* under the biological and social perspective and relate them to Internet-based master-worker task computing

- Employ *reinforcement learning* both on Master and Workers
  - Positive payoffs increase probability of strategy just chosen
  - Negative payoffs reduce the probability
  - Knowledge only of the payoffs received, not of the strategies involved

[Camerer 03,Szepesvari 10]

Our Approach

- We introduce the concept of *evolutionary dynamics* under the biological and social perspective and relate them to Internet-based master-worker task computing

- Employ *reinforcement learning* both on Master and Workers
  - Positive payoffs increase probability of strategy just chosen
  - Negative payoffs reduce the probability
  - Knowledge only of the payoffs received, not of the strategies involved

[Camerer 03,Szepesvari 10]

Our Approach

- We introduce the concept of *evolutionary dynamics* under the biological and social perspective and relate them to Internet-based master-worker task computing

- Employ *reinforcement learning* both on Master and Workers
  - Positive payoffs increase probability of strategy just chosen
  - Negative payoffs reduce the probability
  - Knowledge only of the payoffs received, not of the strategies involved

[Camerer 03,Szepesvari 10]

- Objective: Develop a reliable computation platform where the master obtains the correct task results (whp).

Model: Master

$p_{\mathcal{A}}$

Model: Master

- The Master can audit the responses (with some cost)
  - Auditing means performing the task
  - $p_{\mathcal{A}}$:probability of auditing
    - It may change from round to round

- *Eventual correctness*: After some finite number of rounds, the master obtains the correct task in every round, with minimal auditing, while keeping the workers satisfied

Model: Workers

$$p_{C_i}$$

Model: Workers

- Each worker i has
  - a probability of cheating p_{C_i}
    - It may change from round to round
  - an aspiration a_i
    - the minimum benefit it expects to obtain in a round
- Payofffs

WP_C	Worker's punishment for being caught cheating
WC_T	Worker's cost for computing a task
WB_y	Worker's benefit from master's acceptance

Master's Protocol

Set initial $p_{\mathcal{A}}$ (e.g., 0.5)

Repeat

Send a task to all *n* workers

Upon receiving all answers **do**

Audit the answers with probability $p_{\mathcal{A}}$

If the answers were *not* audited **then**

Accept the value returned by the majority

Else

$$p_{\mathcal{A}} \leftarrow p_{\mathcal{A}} + \alpha_m \cdot \left(\frac{cheaters}{n} - \tau \right)$$

Give appropriate payoff Π_i to each worker i

α_m : learning rate (tunes the extent of change)

τ : tolerance (tolerable ration of cheaters, e.g., 0.5)

Protocol for Worker i

Set initial p_{C_i} (e.g., 0.5)

Repeat

Receive a task from the master

Set S_i = -1*with* probability p_{C_i}, S_i = 1 otherwise

If S_i = 1 then **compute** the task and **send** the result

Else send an arbitrary result

Get payoff Π_i

$$p_{C_i} \leftarrow p_{C_i} - \alpha_w \cdot (\Pi_i - a_i) \cdot S_i$$

α_w: learning rate (tunes the extent of change)

Results

- We analyze the evolution of the master-worker system as a *Markov chain* and we show:

  For the system to achieve eventual correctness, it is necessary and sufficient to set

- *Convergence time*: The number of rounds to achieve eventual correctness
  - We show, both in expectation and with high probability, that our mechanism reaches convergence time quickly
  - We complement the analysis with simulations.

Results

- We analyze the evolution of the master-worker system as a *Markov chain* and we show:

  For the system to achieve eventual correctness, it is necessary and sufficient to set

$$WB_y \geq a_i + WC_T, \quad \forall i \in Z, \; |Z| > n/2$$

- *Convergence time*: The number of rounds to achieve eventual correctness
  - We show, both in expectation and with high probability, that our mechanism reaches convergence time quickly
  - We complement the analysis with simulations.

Results

- We analyze the evolution of the master-worker system as a *Markov chain* and we show:

  For the system to achieve eventual correctness, it is necessary and sufficient to set

$$WB_y \geq a_i + WC_T, \quad \forall i \in Z, \; |Z| > n/2$$

Results

- We analyze the evolution of the master-worker system as a *Markov chain* and we show:

  For the system to achieve eventual correctness, it is necessary and sufficient to set

$$WB_y \geq a_i + WC_T, \quad \forall i \in Z, \; |Z| > n/2$$

Results

- We analyze the evolution of the master-worker system as a *Markov chain* and we show:

  For the system to achieve eventual correctness, it is necessary and sufficient to set

$$WB_y \geq a_i + WC_T, \quad \forall i \in Z, \; |Z| > n/2$$

Results

- We analyze the evolution of the master-worker system as a *Markov chain* and we show:

  For the system to achieve eventual correctness, it is necessary and sufficient to set

$$WB_y \geq a_i + WC_T, \quad \forall i \in Z, \; |Z| > n/2$$

- *Convergence time*: The number of rounds to achieve eventual correctness
  - We show, both in expectation and with high probability, that our mechanism reaches convergence time quickly
  - We complement the analysis with simulations.

Examples of Convergence

- *Under certain conditions, the expected convergence time is*

$$\left(\alpha_w \cdot (WB_y - WC_T - \max_i\{a_i\}) \cdot \varepsilon\right)^{-1}$$

where

$$\varepsilon \in (0, 1 - (WC_T + \max_i\{a_i\})/WB_y).$$

- *Under certain conditions, the converge time is at most*

$$\ln(1/\varepsilon)/p + 1/dec$$

with probability at least

$$(1-\varepsilon)(1-e^{-n/96})(1-e^{-n/36})^{1/dec}$$

where

$$dec = \min_i\{\alpha_w \cdot \min\{a_i,\ WB_y - WC_T - a_i\}\}, \text{and } \varepsilon \in (0,1)$$

Thank you!

Chryssis Georgiou

chryssis@cs.ucy.ac.cy

www.cs.ucy.ac.cy/~chryssis

Internet-based Task Computing

Internet-based Task Computing

- Internet emerges as a viable platform for supercomputing
  - Grid and Cloud computing
    - e.g., EGEE Grid, TERA Grid, Amazon's EC2
  - Master-Worker volunteer computing: @home projects
    - e.g., SETI@home, AIDS@home, Folding@home, PrimeNet

Internet-based Task Computing

- Internet emerges as a viable platform for supercomputing
  - Grid and Cloud computing
    - e.g., EGEE Grid, TERA Grid, Amazon's EC2
  - Master-Worker volunteer computing: @home projects
    - e.g., SETI@home, AIDS@home, Folding@home, PrimeNet

Great potential limited by untrustworthy entities

Top Three Supercomputers (2012)

Top Three Supercomputers (2012)

- **K Computer**, RIKEN Advanced Institute for Computational Science (2011), Japan
  - CPUs: 88,128 SPARC64 VIIIfx 8-core 2.0GHz
  - 11,280 TFLOPS (11.2 PetaFLOPS)

Top Three Supercomputers (2012)

- **K Computer**, RIKEN Advanced Institute for Computational Science (2011), Japan
  - CPUs: 88,128 SPARC64 VIIIfx 8-core 2.0GHz
  - 11,280 TFLOPS (11.2 PetaFLOPS)

Top Three Supercomputers (2012)

- **K Computer**, RIKEN Advanced Institute for Computational Science (2011), Japan
  - CPUs: 88,128 SPARC64 VIIIfx 8-core 2.0GHz
  - 11,280 TFLOPS (11.2 PetaFLOPS)
- **Tianhe-1A,** Tianjin National Supercomputer Center, (2010), China
  - CPUs: 186,368 NUDT X5670 6-core 2.93GHz
  - 4,701 TFLOPS (4.7 PetaFLOPS)

Top Three Supercomputers (2012)

- **K Computer**, RIKEN Advanced Institute for Computational Science (2011), Japan
  - CPUs: 88,128 SPARC64 VIIIfx 8-core 2.0GHz
  - 11,280 TFLOPS (11.2 PetaFLOPS)
- **Tianhe-1A,** Tianjin National Supercomputer Center, (2010), China
  - CPUs: 186,368 NUDT X5670 6-core 2.93GHz
  - 4,701 TFLOPS (4.7 PetaFLOPS)

Top Three Supercomputers (2012)

- **K Computer**, RIKEN Advanced Institute for Computational Science (2011), Japan
  - CPUs: 88,128 SPARC64 VIIIfx 8-core 2.0GHz
  - 11,280 TFLOPS (11.2 PetaFLOPS)
- **Tianhe-1A,** Tianjin National Supercomputer Center, (2010), China
  - CPUs: 186,368 NUDT X5670 6-core 2.93GHz
  - 4,701 TFLOPS (4.7 PetaFLOPS)
- **Jaguar Cray XT5-HE,** Cray Inc (2009), USA
  - CPUs: 224,162 AMD x86_64 Opteron 6-core 2.6GHz

    2,331 TFLOPS (2.33 PetaFLOPS)

Top Three Supercomputers (2012)

- **K Computer**, RIKEN Advanced Institute for Computational Science (2011), Japan
  - CPUs: 88,128 SPARC64 VIIIfx 8-core 2.0GHz
  - 11,280 TFLOPS (11.2 PetaFLOPS)
- **Tianhe-1A,** Tianjin National Supercomputer Center, (2010), China
  - CPUs: 186,368 NUDT X5670 6-core 2.93GHz
  - 4,701 TFLOPS (4.7 PetaFLOPS)
- **Jaguar Cray XT5-HE,** Cray Inc (2009), USA
  - CPUs: 224,162 AMD x86_64 Opteron 6-core 2.6GHz

    2,331 TFLOPS (2.33 PetaFLOPS)

$ $ $ $ $ $

SETI

- Search for ExtraTerrestrial Inteligence

SETI

- Search for ExtraTerrestrial Inteligence
- Internet-based public volunteer computing project
  - Employs the BOINC software platform
  - Hosted by the Space Sciences Laboratory, at the University of California, Berkeley, USA

SETI

- Search for ExtraTerrestrial Inteligence
- Internet-based public volunteer computing project
  - Employs the BOINC software platform
  - Hosted by the Space Sciences Laboratory, at the University of California, Berkeley, USA
- Purpose: analyze radio (telescopic) signals, searching for signs of extra terrestrial intelligence

SETI

- Search for ExtraTerrestrial Inteligence
- Internet-based public volunteer computing project
  - Employs the BOINC software platform
  - Hosted by the Space Sciences Laboratory, at the University of California, Berkeley, USA
- Purpose: analyze radio (telescopic) signals, searching for signs of extra terrestrial intelligence
- How to use it:
  - Register your PC
  - Downloads the SETI data analyzer (screensaver mode)
  - When PC is idling, it starts analyzing data
  - When done, sends results, gets new data chunk to analyze

SETI

- Search for ExtraTerrestrial Inteligence
- Internet-based public volunteer computing project
  - Employs the BOINC software platform
  - Hosted by the Space Sciences Laboratory, at the University of California, Berkeley, USA
- Purpose: analyze radio (telescopic) signals, searching for signs of extra terrestrial intelligence
- How to use it:
  - Register your PC
  - Downloads the SETI data analyzer (screensaver mode)
  - When PC is idling, it starts analyzing data
  - When done, sends results, gets new data chunk to analyze

http://setiathome.berkeley.edu/

SETI@home vs K-Computer

SETI@home vs K-Computer

- **K Computer**, RIKEN Advanced Institute for Computational Science (2011), Japan
  - CPUs: 88,128 SPARC64 VIIIfx 8-core 2.0GHz
  - 11,280 TFLOPS (11.2 PetaFLOPS)

SETI@home vs K-Computer

- **K Computer**, RIKEN Advanced Institute for Computational Science (2011), Japan
  - CPUs: 88,128 SPARC64 VIIIfx 8-core 2.0GHz
  - 11,280 TFLOPS (11.2 PetaFLOPS)

SETI@home vs K-Computer

- **K Computer**, RIKEN Advanced Institute for Computational Science (2011), Japan
  - CPUs: 88,128 SPARC64 VIIIfx 8-core 2.0GHz
  - 11,280 TFLOPS (11.2 PetaFLOPS)

- SETI@home: November 2009
  - 278,832 active CPUs (out of a total of 2.4 million) in 234 countries
  - 769 TFLOPs

SETI@home vs K-Computer

- **K Computer**, RIKEN Advanced Institute for Computational Science (2011), Japan
  - CPUs: 88,128 SPARC64 VIIIfx 8-core 2.0GHz
  - 11,280 TFLOPS (11.2 PetaFLOPS)

- SETI@home: November 2009
  - 278,832 active CPUs (out of a total of 2.4 million) in 234 countries
  - 769 TFLOPs

Comparable processing power with top Supercomputers

@ a fraction of the cost!